UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/749,260	12/31/2003	Dilip Madhusudan Ranade	5760-18700	8564

86942   7590   01/28/2010
Meyertons, Hood, Kivlin, Kowert, Goetzel/Symantec
P.O. Box 398
Austin, TX 78767-0398

EXAMINER
AHN, SANGWOO

ART UNIT	PAPER NUMBER
2168	

NOTIFICATION DATE	DELIVERY MODE
01/28/2010	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

patent_docketing@intprop.com
ptomhkkg@gmail.com

UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

_____

Ex parte DILIP MADHUSUDAN RANADE

_____

Appeal 2009-004179
Application 10/749,260
Technology Center 2100

_____

Decided: January 26, 2010

_____

Before JAMES D. THOMAS, JOHN A. JEFFERY, and THU A. DANG,
Administrative Patent Judges.

JEFFERY, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellant appeals under 35 U.S.C. § 134(a) from the Examiner's
rejection of claims 30-51. We have jurisdiction under 35 U.S.C. § 6(b). We
reverse and enter new grounds of rejection under 37 C.F.R. § 41.50(b).

<div align="center">STATEMENT OF THE CASE</div>

Appellant invented a distributed data object sharing system where data objects are replicated across computing nodes. The system can identify replica conflicts and modify tree structures to reflect the conflicts.[1] Claim 30 is illustrative:

30. A system comprising:

a network; and

a plurality of computing nodes coupled via the network;

wherein the plurality of nodes includes a first node operable to:

create a first file prepresenting a first version of a data object;

detect a conflict between a first replica of the first version of the data object and a second replica of the first version of the data object;

in response to detecting the conflict:

modify a tree structure representing the data object to reflect the conflict, wherein modifying the tree structure comprises adding information to the tree structure representing a branching from the first version of the data object to a second version of the data object and a third version of the data object, wherein the first replica of the first version of the data object represents the second version of the data object and the second replica of the first version of the data object represents the third version of the data object;

create a second file representing the second version of the data object; and

create a third file representing the third version of the data object.

---

[1] *See generally* Abstract; Spec. 3; Fig. 2.

The Examiner relies on the following as evidence of unpatentability:

| Ecklund | US 4,853,843 | Aug. 1, 1989 |
| Tuli | US 6,003,034 | Dec. 14, 1999 |

1. The Examiner rejected claims 30-32, 34-40, 42-45, and 47-50 under 35 U.S.C. § 102(b) as anticipated by Ecklund. Ans. 3-6.[2]

2. The Examiner rejected claims 33, 41, 46, and 51 under 35 U.S.C. § 103(a) as unpatentable over Ecklund and Tuli. Ans. 6-7.

THE ANTICIPATION REJECTION

Regarding the independent claims, the Examiner finds that Ecklund creates a first file representing a first version of a data object, namely (1) the initial set of versions of data objects, and (2) the root of the instantiation tree of a design stored in Ecklund's system. *See* Ans. 4. The Examiner further finds that Ecklund creates second and third files representing respective versions of the data object. *Id.* According to the Examiner, since each version is a data object that "copies" the original version, it could be a stored file. Ans. 4, 8.

Appellant argues that although Ecklund stores data objects in a distributed database, these data objects are not files. Br. 8. As such, Appellant contends that Ecklund does not create three files representing respective data object versions as claimed. *Id.*

---

[2] Throughout this opinion, we refer to the Appeal Brief filed September 5, 2007 and the Examiner's Answer mailed November 27, 2007.

The issue before us, then, is as follows:

ISSUE

Under § 102, has Appellant shown that the Examiner erred in rejecting the independent claims by finding that Ecklund creates three files representing respective data object versions as claimed?

FINDINGS OF FACT

The record supports the following findings of fact (FF) by a preponderance of the evidence:

Appellant's Disclosure

1. Appellant's Specification notes that "[f]or example, in one embodiment a data object may comprise a file. Thus, the data object replicas 109 may comprise replicas of files. In general, a data object may comprise data or information of any kind, where the data is organized or structured in any way." Spec. 8:16-19.

2. Appellant's Specification adds that "[f]or example, in one embodiment, the data object corresponding to a replica version tree may comprise a file, and each replica version may be represented as a corresponding file in a file system." Spec. 18:2-4. *Accord* Spec. 18:7-9.

3. The box corresponding to reference numeral 109 in Figure 2 of the present application is labelled "Data Object Replicas (e.g., File Replicas)." Fig. 2.

3A. "In some [prior art] distributed filed sharing systems, files may be replicated on multiple nodes in the system. Some distributed file sharing systems allow concurrent updates to different replicas in order to improve performance." Spec. 1:28-30.

Ecklund

4. Ecklund's distributed database system provides multiple "virtual partitions" that each store separate instances of an initial database. Ecklund, col. 2, ll. 56-58.

5. Each virtual partition is a collection of sites that (1) have access to a copy of the database, and (2) can communicate with each other. Ecklund, col. 2, ll. 25-30.

6. The database comprises an initial set of versions of data objects. Each version of a particular data object is created by modifying an existing version of the data object. Ecklund, col. 2, ll. 60-63.

7. The initial database includes an initial set of directory data associated with each data object that identifies "paths of descendancy" for the associated data object that comprise sequentially created versions of the data object. Ecklund, col. 2, ll. 63-67.

8. Each virtual partition (1) independently executes group updates including adding a new version of a data object to a path, and (2) maintains a separate change list describing all group updates that it executes. Ecklund, col. 3, ll. 15-24.

9. Each virtual partition may provide a merged database reflecting changes to the initial database resulting from all group updates in accordance with the change list. Ecklund, col. 3, ll. 25-31.

PRINCIPLES OF LAW

Anticipation is established only when a single prior art reference discloses, expressly or under the principles of inherency, each and every element of a claimed invention as well as disclosing structure which is capable of performing the recited functional limitations. *RCA Corp. v. Appl. Dig. Data Sys., Inc.*, 730 F.2d 1440, 1444 (Fed. Cir. 1984); *W.L. Gore & Assoc., Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1554 (Fed. Cir. 1983).

"Inherency . . . may not be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient." *In re Robertson*, 169 F.3d 743, 745 (Fed. Cir. 1999) (citations omitted).

ANALYSIS

This appeal hinges on one fundamental question: Is a *data object* necessarily a *file*? Specifically, the Examiner contends that Ecklund's *data objects* anticipate the three recited *files* representing the respective three versions of the data object as claimed. Ans. 4, 8. Appellant, however, argues that although Ecklund stores data objects in a distributed database, these data objects are not files. Br. 8. We agree with Appellant to the extent that while Ecklund's data objects *may* be files, they are not *necessarily* files—a crucial requirement for anticipation.

Appellant's Specification confirms this point. Specifically, the Specification notes that, in *one exemplary embodiment*, "a data object *may* comprise a file." FF 1 (emphases added). Similarly, the Specification indicates that, in *one exemplary embodiment*, "the data object corresponding

to a replica version tree *may* comprise a file, and each replica version *may* be represented as a corresponding file in a file system." FF 2 (emphases added).

Appellant's permissive word choice here is telling, and confirms that while a data object *may* be a file (e.g., in one embodiment (FF 1-2)), it is not so limited. That Appellant repeatedly indicates that file-type data object is *exemplary* in connection with data object replicas (FF 1-3) only bolsters this conclusion.

The Examiner, however, takes the position that "a data object stored in a database *could* clearly be a file." Ans. 8 (emphasis added). While this may be true, a data object is not *necessarily* a file. That is, the record before us indicates that files are merely types of data objects. Therefore, even assuming that a file is a data object stored in some type of database as the Examiner asserts (*Id.*), the converse is not necessarily true. That is, all files are data objects, but not all data objects are necessarily files. The Examiner points to nothing on this record indicating the contrary.

And this distinction is the very reason why the Examiner's anticipation rejection is untenable. To be sure, Ecklund's database system creates different versions of data objects. FF 6-7. And "virtual partitions" in Ecklund independently execute group updates including adding new versions of data objects to paths, and providing merged databases reflecting changes to the initial database in accordance with these updates. FF 8-9.

But the Examiner has not shown that these data objects are *necessarily* files. That these data objects "could be" files as the Examiner speculates (Ans. 8) is not enough. To anticipate, Ecklund's data objects must *necessarily* be files—which they are not. *See Robertson*, 169 F.3d at 745.

For these reasons, Appellant has persuaded us of error in the Examiner's anticipation rejection of independent claims 30, 38, 43, and 48. We therefore reverse the Examiner's rejection of these claims, and dependent claims 31, 32, 34-37, 39, 40, 42, 44, 45, 47, 49, and 50 for similar reasons.

THE OBVIOUSNESS REJECTION

Since the Examiner has not shown that Tuli cures the deficiencies noted above regarding the independent claims, we will not sustain the obviousness rejection of dependent claims 33, 41, 46, and 51 for similar reasons.

NEW GROUND OF REJECTION UNDER 37 C.F.R. § 41.50(B)

Claims 30, 38, 43, and 48 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Ecklund.[3] Regarding these independent claims, Appellant does not dispute the Examiner's findings regarding the disclosure to Ecklund (Ans. 3 and 6), except with respect to Ecklund's storage of data objects as not corresponding to a file. *See* Br. 7-8. Although data objects

---

[3] Although we decline to reject every claim under our discretionary authority under 37 C.F.R. 41.50(b), we emphasize that our decision does not mean the remaining claims are patentable. Rather, we leave the patentability determination of these claims to the Examiner. *See* MPEP § 1213.02.

are not necessarily files as we indicated previously, they nevertheless can be files—a fact readily acknowledged in the Specification. *See* FF 1-3. Furthermore, Appellant acknowledges in the Specification that file replication on multiple nodes is well known in distributed file sharing systems. FF 3A.

Since files are well known types of data objects, and Ecklund stores multiple data objects in a distributed database associated with virtual partitions (*see* FF 1-9), it would have been obvious to one of ordinary skill in the art at the time of the invention to create three files representing respective data object versions in lieu of data objects for such a purpose. Such a modification is tantamount to the predictable use of prior art elements according to their established functions. *See KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 417 (2007).

CONCLUSION

Appellant has shown that the Examiner erred in rejecting (1) claims 30-32, 34-40, 42-45, and 47-50 under § 102, and (2) claims 33, 41, 46, and 51 under § 103. We have, however, entered new grounds of rejection for claims 30, 38, 43, and 48.

This decision contains new grounds of rejection pursuant to 37 C.F.R. § 41.50(b) that provides that "[a] new ground of rejection . . . shall not be considered final for judicial review."

Section 41.50(b) also provides that Appellant, WITHIN TWO MONTHS FROM THE DATE OF THE DECISION, must exercise one of the following two options with respect to the new ground of rejection to avoid termination of the appeal as to the rejected claims:

> (1) Submit an appropriate amendment of the claims so rejected
> or new evidence relating to the claims so rejected, or both, and
> have the matter reconsidered by the examiner, in which event
> the proceeding will be remanded to the examiner. . . .
>
> (2) Request that the proceeding be reheard under § 41.52 by the
> Board upon the same record. . . .

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

<u>REVERSED</u>
<u>37 C.F.R. § 41.50(b)</u>

pgc

Meyertons, Hood, Kivlin, Kowert, Goetzel/Symantec
P.O. Box 398
Austin, TX 78767-0398